

## **AMENDMENTS TO THE SPECIFICATION**

*Please replace paragraph [0006] with the following marked-up version of the paragraph:*

[0006] Regardless of the protocol used, however, it may be desirable to extend the data properties and functionalities of a particular messaging board protocol. For example, some protocols are unsecured with no way of authenticating a user or, like NNTP, offer limited security options. With no way to authenticate or provide for a secure connection, there is no true reliability in the identity or status of an author of a message nor is there any way to protect data from being tampered with. Further, messaging board protocols are static in that data properties associated with the posting cannot easily be updated or ~~dynamic~~ dynamically changed. This is due to various reasons, such as the difficulty in getting a change through a standards committee. As mentioned in greater detail below, however, even if you can get a change approved, significant changes in the protocol will likely break legacy clients that don't support these changes. Still, there are a number of other reasons for extending messaging board protocols.

*Please replace paragraph [0009] with the following marked-up version of the paragraph:*

[0009] One problem associated with the current thread structures is that a user must browse through much of a thread in order to determine whether the thread represents an ongoing conversation or a conversation that has ended. For example, some companies employ software experts who monitor and respond to questions posted on their message board server. With traditional message board protocols and clients, the expert must go through each conversation thread to determine if there is a question, and whether there has been an appropriate answer to that question. As can be appreciated, this becomes a time- and labor-intensive process. As such, it is desirable to be able to identify messages ~~message~~ threads based on some user-specified criteria, like whether a question has been sufficiently answered. Again, however, such functionality would also require the extension of typical message board protocols.

*Please replace paragraph [0041] with the following marked-up version of the paragraph:*

[0041] The above mentioned updated information example embodiment may be optimized for greatest efficiency and scalability. For example, a client 110 when requesting for updated information from server 105 may potentially send a timestamp indicating the client's most recently received updated metadata. Accordingly, the server 105 can evaluate the extended data including the metadata that has subsequently been updated, and thus send only that extended data that has changed. Of course, other well [[know]] known methods for optimizing downloading efficiency can also be utilized, and therefore the above example is used for illustrative purposes only and is not meant to limit or otherwise narrow the scope of the present application. Although other legacy protocols, such as NNTP, used timestamps to determine what messages had previously been received, there was no evaluation of what extended data had changed because these legacy protocols didn't support such data properties.

*Please replace paragraph [0046] with the following marked-up version of the paragraph:*

[0046] As an alternative to thread re-typing, the user-specified message criteria may be to show only unanswered threads. Accordingly, this embodiment would allow for a minimal visual representation 315 collapsing the entire thread 310 to display [[a]] nothing. As such, the term minimal visual representation is meant to include the null display.

*Please replace paragraph [0056] with the following marked-up version of the paragraph:*

[0056] Other example embodiments provide for a method of visually collapsing messages within a message thread based upon user-specified criteria for improving the user experience and efficiency of downloading a message. The method may include the acts of receiving an initial message from a client for posting on a message board server. The initial message having extended attributes that include a message type, visual representation data, and data identifying the type of person posting the initial message. Further, the method may include the act of receiving one [[more]] or more subsequent messages from one or more clients for posting on a message board server. Similar to the initial message, the one or more subsequent messages may have extended attributes that include a message type, visual representation data, and data identifying the type of person posting the one or more messages. Further, the method may

include the act of correlating the initial message with the one or more subsequent messages for creating a message thread, which is a list of the visual representation data for the initial message and the one or more subsequent messages. When determining if the message thread can be collapsed, the method may include the act of receiving user-specified message criteria important to a user. Based on the user-specified message criteria, an act of evaluating one or more of the extended attributes in the one or more subsequent messages against one or more of the extended attributes in the initial message is provided for determining if the message thread can be collapsed into a minimized visual representation of the message tree.

*Please replace paragraph [0057] with the following marked-up version of the paragraph:*

[0057] The minimized visual representation can possibly be simply a non display of any information about the thread. Further, the message type for the initial message and the one or more subsequent messages can be chosen from one of a question, comment, suggestion, answer, an answered question or an answered suggestion. Moreover, the extended attributes for the initial message and the one or more subsequent messages may further include profiler data, which has details about a user who posted the message including at least one of the user's integrity or how long the user has been posting messages to the message board server. In addition, the extended attributes for the initial message and the one or more subsequent messages may further include extended data properties that are periodically updated based on voting input from one or more users, and wherein a client periodically makes a request for the extended data properties that have changed. The voting input from the one or more users may be an opinion rating the message as one or more of useful, not useful, useful but need more information, specifically the answer to an original question, or spam.